Grid Computing

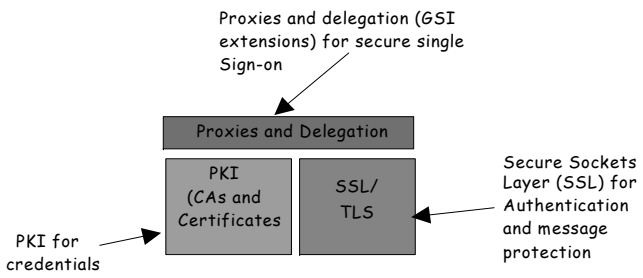
Security - GSI

Paul A. Farrell
Fall 2006

Grid Security Infrastructure (GSI)

- Set of tools, libraries, and protocols used in Globus and other grid middleware, to allow users and applications to securely access resources securely
- Based on PKI, with certificate authorities and X509 certificates.
- GSI provides:
  - A public-key system;
  - Mutual authentication through digital certificates;
  - Credential delegation and single sign-on.
- Motivated by
  - Need for secure, authenticated communication in the Grid
  - Need to support security across organizations, but without central management
  - Need to support single sign-on, including delegation of credentials for computations that involve multiple resources and/or sites

Grid Security Infrastructure (GSI)



Basics

- Public-key cryptography
  - Two keys, private and public, one to encrypt, one to decrypt
- Digital signature
  - A hash of the message, encrypted with my private key
- Certificate
  - My public key, digitally signed by the Certificate Authority
  - IF you trust the CA, AND believe that you have the public key of the CA, THEN you can believe that the public key in the message is mine
- Mutual authentication
  - If two parties have certificates, and both parties trust the CA that signed the other's certificate, then the two parties can prove to each other that they are who they say they are
- Confidential communication
  - Encrypted communication is NOT the default in GSI. However, the public keys can be used to exchange a shared secret key for encrypted messages if desired

Grid security technologies & requirements

- Technologies
  - Security is NOT based on interorganizational trust relationships
  - It is based on the use of a virtual organization (VO) as a bridge among entities in a particular community or function
- Requirements
  - Must support scalable, dynamic, distributed VO's
  - Key attributes of VO's is that
    - Participants and resources are governed by classical organizations of which they are members
    - Some VO's are long-lived, other short lived, so the overhead of security must be small
  - VO access must be established and coordinated
    - Between the local user and the organization
    - Between the VO and the user
  - CANNOT assume trust relationships between the classical organization and the VO or its external members

Paul A. Farrell 2006 KENT STATE Grid Computing 5

Acquiring certificates

All users and services need to have a certificate issued from a trusted certificate authority (CA).

- It is highly recommended that the builders of production grids either establish their own trusted CA or use an established commercial CA.
- The SimpleCA package can be used to run your own CA.

Paul A. Farrell 2006 KENT STATE Grid Computing 6

Acquiring user and host certificates

- To request a user certificate, the user simply runs "grid-cert-request" on a system that has GT4 installed.
- grid-cert-request will ask for a password to protect your key, and give you a set of instructions for how to mail your request to the CA.
- To request a host certificate, the administrator (root) runs "grid-cert-request"

Paul A. Farrell 2006 KENT STATE Grid Computing 7

grid-cert-request

- When you run the grid-cert-request command, it will generate three files.
  - usercert_request.pem: the request that you need to send to the CA
  - userkey.pem: contains the private key
  - usercert.pem, which will be a 0 byte file. *This is not your certificate!* It is merely a placeholder that helps to remind you where to put your certificate when the CA responds to your request.

Paul A. Farrell 2006 KENT STATE Grid Computing 8

User and host certificates

- First double-check the subject.
- Send the request to your Certificate Authority via email.
- When you retrieve your certificate, save it to ~/.globus/usercert.pem.
- Do not lose this file, and do not forget your password.

- A similar process is followed by root for a host certificate
- The files are stored in **/etc/grid-security/**
- When retrieved the signed certificate is stored in **/etc/grid-security/hostcert.pem**

Securing private keys

- Core GSI does not store private keys in an encrypted format
  - It relies on the protection mechanism of the local file system to store private keys
- On linux, make sure that the private key userkey.pem is only readable by you i.e has permissions r--------

GSI client-side authorization

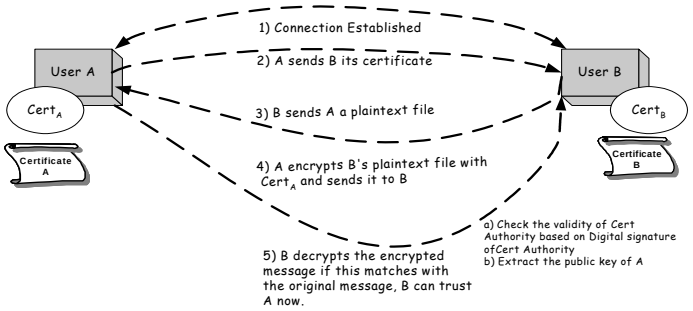
A client has every right to be picky about what services it accesses, but it is not common

- None – No authorization will be performed
- Self – the will authorize an invocation if the service's identity is the same as the client's (based on X.509 subject in the certificate or proxy).
- Host – the client will authorize an invocation if the host returns an identity containing the hostname

GSI server side authorization

- None – no authorization is needed to use this service
- Self – client will be allowed to use a grid service if the client's identity is the same as the service's identity
- Gridmap – a list of 'authorized users' akin to an ACL (Access Control List).

Mutual Authentication



B send its certificate and A then authenticates it similarly

Delegation and single sign-on

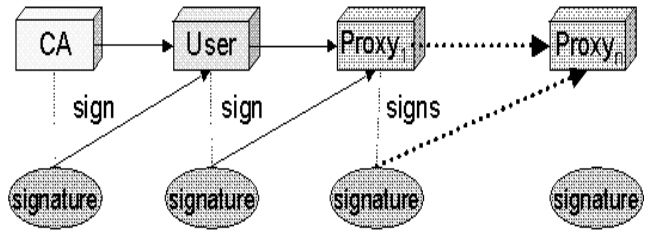
- If a Grid computation requires that several Grid resources be used (each requiring mutual authentication), or if there is a need to have agents (local or remote) requesting services on behalf of a user, the need to re-enter the user's passphrase can be avoided by creating a *proxy*.

Proxy certificate

A proxy consists of

- A new certificate (with a new public key in it) and a new private key.
- The new certificate contains the owner's identity, modified slightly to indicate that it is a proxy.
- The new certificate is signed by the owner, rather than a CA.
- The certificate also includes a time notation after which the proxy should no longer be accepted by others.
- The proxy certificate is stored in /tmp on linux usually as /tmp/x509up_u*userID* or /tmp/x509up_u_*username*

Use of proxy certificate



Use of proxy certificate

- The proxy's private key must be kept secure, but not for very long
- Usually the proxy's private key is kept in a local storage system without being encrypted
  - File permissions prevent anyone else from looking at them easily.
- Once a proxy is created and stored, the user can use the proxy certificate and private key for mutual authentication without entering a password.

Proxy certificate chain of trust

- The remote party receives the proxy's certificate (signed by the owner), and also the owner's certificate.
- During mutual authentication, the owner's public key (obtained from the certificate) is used to validate the signature on the proxy certificate.
- The CA's public key is then used to validate the signature on the owner's certificate.
- This establishes a chain of trust from the CA to the proxy through the owner.

Creating a proxy certificate

- To create a proxy with the default expiration (12 hours), run the grid-proxy-init program. For example:
  % grid-proxy-init
- The subject of a proxy certificate is the same as the subject of the certificate that signed it, with **/CN=proxy** added to the name.
- A host gatekeeper will accept job requests submitted by the user, as well as any proxies he has created.

GSI system configuration

- GSI Directories
  - Trusted CA directory: contains the CA certificates and associated files trusted by the globus installation
  - If they are generated using SimpleCA by user globus they are in ~globus/.globus/SimpleCA
  - Grid Security directory: contains symbolic links to the certificate request configuration files

grid-mapfile

- A file containing entries mapping X.509 certificate subjects to local user names.
- This file can also serve as an access control list for GSI enabled services
  - But can still use GGF SAML Callout to other AuthZ functions
- Is typically found in
  /etc/grid-security/grid-mapfile.

Grid Security Infrastructure (GSI)

Example:

```
"/C=UK/O=eScience/OU=Glasgow/L=Compserv/CN=steve kee" skee
"/C=UK/O=eScience/OU=Edinburgh/L=NeSC/CN=stewart mills" smills
"/C=UK/O=eScience/OU=Glasgow/L=Compserv/CN=iain mcbride"
   imcbride
"/C=UK/O=eScience/OU=Aberdeen/L=GeSC/CN=nikki salter" nsalter
"/C=UK/O=eScience/OU=Newcastle/L=NEReSC/CN=nicola wightman"
   nwightman
"/C=UK/O=eScience/OU=London/L=LeSC/CN=scott mccaig" smccaig
"/C=UK/O=eScience/OU=Glasgow/L=Compserv/CN=kev mcneil" kmcneil
"/C=UK/O=eScience/OU=Glasgow/L=Compserv/CN=nik martin" nmartin
"/C=UK/O=eScience/OU=Edinburgh/L=NeSC/CN=ann robertson"
   aroberts
```


Federated Trust

- Local authentication infrastructures are vital
  - e.g. Campus student directories
    - Support existing infrastructures (e.g. registration, human resources)
      - Will normally have enrolled IN PERSON at the institution
        - » With standard identity (birth certificate, exam results)
      - Will be (reasonably) well known by local staff
  - Also the Regional Operators for a CA
    - Required decentralisation of credential verification due to travel/time restrictions
      - National CA would be impossible without this
- Remote authentication information will always be out of date
  - Don't want to have to learn lots of usernames/passwords

Federated Trust

- The best entity to authenticate a person is their home institution/company
  - Info will be up to date
  - They will always know a person better than a remote site
  - Remote site may not know if user is still valid or not
- Can we utilise a user's home credentials to access remote resources?